```
UUU        UUU  EEEEEEEEEEEEEE  TTTTTTTTTTTTTT  PPPPPPPPPPP       SSSSSSSSSSS  YYY            YYY
UUU        UUU  EEEEEEEEEEEEEE  TTTTTTTTTTTTTT  PPPPPPPPPPP       SSSSSSSSSSS  YYY            YYY
UUU        UUU  EEEEEEEEEEEEEE  TTTTTTTTTTTTTT  PPPPPPPPPP        SSSSSSSSSSS  YYY            YYY
UUU        UUU  EEE                  TTT        PPP      PPP  SSS              YYY            YYY
UUU        UUU  EEE                  TTT        PPP      PPP  SSS              YYY            YYY
UUU        UUU  EEE                  TTT        PPP      PPP  SSS              YYY            YYY
UUU        UUU  EEE                  TTT        PPP      PPP  SSS                  YYY    YYY
UUU        UUU  EEE                  TTT        PPP      PPP  SSS                  YYY    YYY
UUU        UUU  EEE                  TTT        PPP      PPP  SSS                  YYY    YYY
UUU        UUU  EEEEEEEEEEE          TTT        PPPPPPPPPPP       SSSSSSSSS            YYY
UUU        UUU  EEEEEEEEEEE          TTT        PPPPPPPPPPP       SSSSSSSSS            YYY
UUU        UUU  EEEEEEEEEEE          TTT        PPPPPPPPPPP       SSSSSSSSS            YYY
UUU        UUU  EEE                  TTT        PPP                      SSS           YYY
UUU        UUU  EEE                  TTT        PPP                      SSS           YYY
UUU        UUU  EEE                  TTT        PPP                      SSS           YYY
UUU        UUU  EEE                  TTT        PPP                      SSS           YYY
UUU        UUU  EEE                  TTT        PPP                      SSS           YYY
UUUUUUUUUUUUUU  EEEEEEEEEEEEEE       TTT        PPP              SSSSSSSSSSS           YYY
UUUUUUUUUUUUUU  EEEEEEEEEEEEEE       TTT        PPP              SSSSSSSSSSS           YYY
UUUUUUUUUUUUUU  EEEEEEEEEEEEEE       TTT        PPP              SSSSSSSSSSS           YYY
```

SATSSF08

LIS

SATSSF08
Table of contents

SATSSF08
V04-000

J 3
- SATS SYSTEM SERVICE TESTS (FAILING S. 16-SEP-1984 00:37:27 VAX/VMS Macro V04-00   Page 1
                                        5-SEP-1984 04:28:22 [UETPSY.SRC]SATSSF08.MAR;1   (1)

```
0000    1              .TITLE  SATSSF08 - SATS SYSTEM SERVICE TESTS  (FAILING S.C.)
0000    2              .IDENT  'V04-000'
0000    3
0000    4
0000    5      ;***************************************************************
0000    6      ;*                                                             *
0000    7      ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                    *
0000    8      ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.     *
0000    9      ;*  ALL RIGHTS RESERVED.                                       *
0000   10      ;*                                                             *
0000   11      ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
0000   12      ;*  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *
0000   13      ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
0000   14      ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
0000   15      ;*  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
0000   16      ;*  TRANSFERRED.                                               *
0000   17      ;*                                                             *
0000   18      ;*  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
0000   19      ;*  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
0000   20      ;*  CORPORATION.                                               *
0000   21      ;*                                                             *
0000   22      ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
0000   23      ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.    *
0000   24      ;*                                                             *
0000   25      ;*                                                             *
0000   26      ;***************************************************************
0000   27      ;
0000   28
0000   29      ;++
0000   30      ; FACILITY:      SATS SYSTEM SERVICE TESTS
0000   31      ;
0000   32      ; ABSTRACT:      THE SATSSF08 MODULE TESTS THE EXECUTION OF CERTAIN
0000   33      ; VMS SYSTEM SERVICES, INVOKED IN SUCH A WAY AS TO EXPECT FAILING
0000   34      ; STATUS CODES. THE SYSTEM SERVICES TESTED AND THE STATUS CODES
0000   35      ; EXPECTED ARE SUMMARIZED AS ARGUMENTS TO THE TESTSERV MACROS
0000   36      ; WHICH APPEAR NEAR THE END OF THIS LISTING. SUCCESSFUL STATUS
0000   37      ; CODES ARE TESTED IN OTHER MODULES.
0000   38      ;
0000   39
0000   40      ; ENVIRONMENT: USER MODE IMAGE; NEEDS CMKRNL PRIVILEGE,
0000   41      ;              DYNAMICALLY ACQUIRES OTHER PRIVILEGES, AS NEEDED.
0000   42
0000   43      ; AUTHOR: THOMAS L. CAFARELLA,          CREATION DATE: MAY, 1977
0000   44      ;         PAUL D. FAY (DISPSERV & TESTSERV MACROS)
0000   45
0000   46      ; MODIFIED BY:
0000   47      ;
0000   48      ;
0000   49      ;     V03-001 RNP0001         Robert N. Perron        07-Oct-1981
0000   50      ;             Changed to reflect change in CMKRNL privilege. CMKRNL
0000   51      ;             now overlaps CMEXEC.
0000   52      ;
0000   53      ;**
0000   54      ;--
```

```
                          0000      56              .SBTTL  DECLARATIONS
                          0000      57  ;
                          0000      58  ; INCLUDE FILES:
                          0000      59  ;
                          0000      60              $PHDDEF                           ; PROCESS HEADER OFFSET SYMBOLS
                          0000      61              $PCBDEF                           ; PROCESS CONTROL BLOCK OFFSET SYMBS
                          0000      62              $STSDEF                           ; STATUS MESSAGE SYMBOLS
                          0000      63              $PRVDEF                           ; SYMBOL DEFS FOR PRIVILEGES
                          0000      64              $UETPDEF                          ; UETP MSG CODE DEFINITIONS
                          0000      65              $SHR_MESSAGES UETP,116,<<TEXT,INFO>>
                          0000      66                                               ; DEFINE UETP$_TEXT
                          0000      67                                               ; GET RID OF MACRO DEFINITIONS
                          0000      68  ;
                          0000      69  ; MACROS:
                          0000      70  ;
                          0000      71  ;
                          0000      72  ; EQUATED SYMBOLS:
                          0000      73  ;
          00000000        0000      74  WARNING         = 0                          ; WARNING SEVERITY VALUE FOR MSGS
          00000001        0000      75  SUCCESS         = 1                          ; SUCCESS SEVERITY VALUE FOR MSGS
          00000002        0000      76  ERROR           = 2                          ; ERROR SEVERITY VALUE FOR MSGS
          00000003        0000      77  INFO            = 3                          ; INFORMATIONAL SEV VALUE FOR MSGS
          00000004        0000      78  SEVERE          = 4                          ; SEVERE (FATAL) SEV VALUE FOR MSGS
          00000000        0000      79  TCG_NO          = 0                          ; INITIALIZE TEST CASE GROUP NUMBER
          00000000        0000      80  GRP_TOTAL       = 0                          ; INITIALIZE TEST CASE GROUP TOTAL
          00007FFF        0000      81  R0_THRU_SP      = ^M<R0,R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,AP,FP,SP>
          00000000        0000      82  ARGLST_CME      = 0                          ; ARGLST ARG FOR CMEXEC ...
                          0000      83                                               ; ... (MISSING ARG LIST)
                          0000      84  ;
                          0000      85  ; OWN STORAGE:
                          0000      86  ;
```

SATSSF08
V04-000

L 3
- SATS SYSTEM SERVICE TESTS (FAILING S. 16-SEP-1984 00:37:27 VAX/VMS Macro V04-00   Page 3
DECLARATIONS   5-SEP-1984 04:28:22 [UETPSY.SRC]SATSSF08.MAR;1   (1)

```
                         00000000          88              .PSECT  RODATA,RD,NOWRT,NOEXE,LONG
                    BFFC 0000          89 REG_COMP_MASK:    .WORD   ^M<R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,AP,FP> ! ^X8000 -
                         0002          90                                            ; REG COMPARE MASK (HIGH-ORDER ...
                         0002          91                                            ; ... BIT MUST BE ON
                         0002          92 ERR_MSG_FAOCTL:   STRING  I,<!/!AC!1ZB!1ZB: REGISTER !2UW CONTENTS ALTERED>, -
                         0002          93                          <;   BEFORE SERVICE CALL: !8XL    AFTER SERVICE CALL: !8XL>
                         006E          94 TEST_MOD_NAME:    STRING  C,<SATSSF08>     ; TEST MODULE NAME
                         0077          95 TEST_MOD_BEG:     STRING  C,<begun>        ; DISPOSITION FIELD OF TEST MOD MSG
                         007D          96 TEST_MOD_SUCC:    STRING  C,<successful>   ; DISPOSITION FIELD OF TEST MOD MSG
                         0088          97 TEST_MOD_FAIL:    STRING  C,<failed>       ; DISPOSITION FIELD OF TEST MOD MSG
                         008F          98 TEST_MOD_NAME_D:  STRING  I,<SATSSF08>     ; TEST MODULE NAME DESCRIPTOR
                         009F          99 TTNAME:           STRING  I,<TT>           ; TERMINAL LOGICAL NAME
   00000000'00000000'    00A9         100 INADR:            .LONG   NOACCESS,NOACCESS ; PAGE ADDRESS OF NOACCESS PSECT
            00000000'    00B1         101 PROT:             .LONG   PRT$C_NA         ; PROTECTION CODE FOR NOACCESS PSECT
   FFFFFFFF FFFFFFFF     00B5         102 ONES:             .LONG   -1,-1            ; A QUADWORD OF 1-BITS
                         00BD         103 ROUTIN_CME:                                ; ROUTIN ARGUMENT FOR CMEXEC
                         00BD         104 ROUTIN_CME10:                              ; ROUTIN ARGUMENT FOR CMEXEC
            0000         00BD         105                   .WORD   0                ; ENTRY MASK FOR CHANGE MODE SERVS
   50    00'8F  9A       00BF         106                   MOVZBL  #SS$_NORMAL,R0   ; R0 LOAD FOR CHANGE MODE SERVICES
                 04      00C3         107                   RET                      ; RETURN INSTR FOR CHANGE MODE SERVS
            00000000     00C4         108 MSGID_GTM:        .LONG   0                ; MSGID ARGUMENT FOR GETMSG
            0000000F     00C8         109 FLAGS_GTM:        .LONG   ^B1111           ; FLAGS ARGUMENT FOR GETMSG
            0FFFFFF0     00CC         110 MSGID_GTM10:      .LONG   ^X0FFFFFF0       ; MSGID ARGUMENT FOR GETMSG
```

SATSSF08
V04-000

M 3
- SATS SYSTEM SERVICE TESTS (FAILING S. 16-SEP-1984 00:37:27 VAX/VMS Macro V04-00   Page   4
DECLARATIONS   5-SEP-1984 04:28:22 [UETPSY.SRC]SATSSF08.MAR;1   (1)

```
        00000000    112         .PSECT  RWDATA,RD,WRT,NOEXE
00000004 0000    113 TPID:         .BLKL   1          ; PROCESS ID FOR THIS PROCESS
00000008 0004    114 CURRENT_TC:   .BLKL   1          ; PTR TO CURRENT TEST CASE
00000044 0008    115 REG_SAVE_AREA: .BLKL  15         ; SAVE AREA FOR ALL REGS (SANS PC)
007480D9 0044    116 MOD_MSG_CODE: .LONG   UETP$_SATSMS ; TEST MODULE MSG CODE FOR PUTMSG
0000004C 0048    117 CLOB_REG_NO:  .BLKL   1          ; CLOBBERED REG NO (FOR FAO ERR MSG)
00000050 004C    118 REG_BEFORE_SS: .BLKL  1          ; REG CONTENTS BEFORE S.S.
         0050    119                                  ; ... (FOR FAO ERROR MSG)
00000054 0050    120 REG_AFTER_SS: .BLKL   1          ; REG CONTENTS AFTER S.S.
         0054    121                                  ; ... (FOR FAO ERROR MSG)
         0054    122 $$TSTN$$:      STRING  C,<  SF  > ; ASCII PORTION OF TEST CASE NAME
0000006E' 005C   123 TMN_ADDR:     .ADDRESS TEST_MOD_NAME ; ADDR OF TEST MOD NAME FOR FAO
00000077' 0060   124 TMD_ADDR:     .ADDRESS TEST_MOD_BEG  ; ADDR OF T.M. DISP FIELD FOR FAO
00000068 0064    125 TS_EP:        .BLKL   1          ; ENTRY PNT FOR CURR TESTSERV MACRO
00000070 0068    126 RETADR:       .BLKL   2          ; RETURN LONGWORDS FOR SETPRT
00000071 0070    127 PRVPRT:       .BLKB   1          ; PROT RETURN BYTE FOR SETPRT
00000079 0071    128 PRIVMASK:     .BLKQ   1          ; ADDR OF PRIVILEGE MASK (IN PHD)
0000007D 0079    129 CHM_CONT:     .BLKL   1          ; CHANGE MODE CONTINUE ADDRESS
00000091 007D    130 REGS:         .BLKL   5          ; AREA FOR COND INDEX REGS (R2-R6)
00000093 0091    131 MSGLEN_GTM:   .BLKW   1          ; MSGLEN ARGUMENT FOR GETMSG
         0093    132 BUFADR_GTM:    STRING  0,256     ; BUFADR ARGUMENT FOR GETMSG
0000019F 019B    133 OUTADR_GTM:   .BLKB   4          ; OUTADR ARGUMENT FOR GETMSG
         019F    134 BUFADR_GTM30:                    ; BUFADR ARGUMENT FOR GETMSG
00000000 019F    135                .LONG   0         ; ZERO LENGTH STRING ...
000001A3' 01A3   136                .ADDRESS .        ; .... AT AN ACCESSIBLE LOCATION
         01A7    137 BUFADR_GTM31:   STRING  0,1      ; BUFADR ARGUMENT FOR GETMSG
```

```
            00000000      139                .PSECT  SATS_ACCVIO_1,RD,WRT,NOEXE,PAGE
            00000200 0000  140 EMPTY:         .BLKB   512     ; RESERVE A PAGE OF SPACE
                     0200  141 ;
                     0200  142 ;+
                     0200  143 ;*****************************************************************************
                     0200  144 ;*                                                                          *
                     0200  145 ;*       THE ORDER OF STATEMENTS IN THIS PSECT IS CRITICAL.                  *
                     0200  146 ;*       DO NOT RE-ARRANGE THE VARIABLES. CONSULT SATS                       *
                     0200  147 ;*       FUNCTIONAL SPECIFICATION FOR A DESCRIPTION OF THE USE               *
                     0200  148 ;*       OF THE EMPTY PSECT (AND ITS COMPANION PSECT, NOACCESS).             *
                     0200  149 ;*                                                                          *
                     0200  150 ;*****************************************************************************
                     0200  151 ;-
                     0200  152 ;
                     0200  153 ; TYPE AAAAA_SSSX1 (TYPE AAAAA_SSSX2 IF NOT DESC) GO HERE:
            000001F3 0200  154                . = . - 13      ; ALLOW ROOM FOR STRING DESCRIPTOR
                     01F3  155 ; TYPE AAAAA_SSSX5 GO HERE:
            00000006 01F3  156                .LONG   6       ; STRING LENGTH (WILL CROSS PSECT BOUNDARY)
            000001FB' 01F7 157                .ADDRESS .+4    ; STRING ADDRESS
                     01FB  158 ; TYPE AAAAA_SSSX3 GO HERE:
            000001FC 01FB  159                .BLKB   1       ; LOW-ORDER BYTE OF STRING LENGTH
                     01FC  160 ; TYPE AAAAA_SSSX2 GO HERE:
            00000200 01FC  161                .BLKL   1       ; STRING LENGTH
                     0200  162 ;
                     0200  163 ;
                     0200  164 ;
                     0200  165 ;
            00000000      166                .PSECT  SATS_ACCVIO_2,RD,WRT,NOEXE,PAGE
            00000200 0000  167 NOACCESS:      .BLKB   512     ; RESERVE A PAGE OF SPACE
            00000000 0200  168                . = . - 512     ; RETURN LOC CTR TO BEGINNING OF PSECT
            00000000' 0000 169                .ADDRESS EMPTY  ; ADDRESS OF ACCESSIBLE STRING
            00000000' 0004 170                .ADDRESS EMPTY/^X100 ; ADDRESS OF ACCESSIBLE STRING
                     0008  171 ;+
                     0008  172 ; *** NOTE -- DO NOT CHANGE LOCATION OR SEQUENCE OF ABOVE STATEMENTS!
                     0008  173 ; ***        THIS PSECT (NOACCESS) MUST APPEAR IN MEMORY IMMEDIATELY
                     0008  174 ; ***        FOLLOWING THE EMPTY PSECT. PSECT NAMES AND OPTIONS WILL BE
                     0008  175 ; ***        CHOSEN TO FORCE THE DESIRED PSECT ORDERING.
                     0008  176 ;-
                     0008  177 ;
                     0008  178 ;
                     0008  179 ;
                     0008  180 ;
            00000000      181                .PSECT  SATSSF08,RD,WRT,EXE,LONG
```

```
0000    183              .SBTTL  SATSSF08
0000    184  ;++
0000    185  ; FUNCTIONAL DESCRIPTION:
0000    186  ;
0000    187  ;            AFTER PERFORMING SOME INITIAL HOUSEKEEPING, SUCH AS
0000    188  ; PRINTING THE MODULE BEGIN MESSAGE AND ACQUIRING ALL PRIVILEGES,
0000    189  ; THE SATSSF08 ROUTINE EXECUTES THE TEST SERV EXEC MACRO TO RUN
0000    190  ; ALL TEST CASES. WHEN THE MACRO COMPLETES ITS EXECUTION, SATSSF08
0000    191  ; PRINTS A TEST MODULE SUCCESS OR FAIL MESSAGE AND EXITS TO THE
0000    192  ; OPERATING SYSTEM. TEST SERV EXEC CALLS THE TC CONTROL/TESTSERV
0000    193  ; CO-ROUTINE PAIR ONCE PER TEST CASE GROUP TO EXECUTE ALL TEST
0000    194  ; CASES IN THAT GROUP. EACH TEST CASE GROUP IS DEFINED BY BOUNDING
0000    195  ; ITS TEST CASES WITH A TC GROUP MACRO BEFORE THE FIRST TEST CASE
0000    196  ; AND A TCEND MACRO AFTER THE LAST ONE. THE TEST CASES THEMSELVES
0000    197  ; ARE DEFINED WITHIN THESE BOUNDS BY PRECEDING EACH WITH A
0000    198  ; NEXT TEST-CASE MACRO. TC-CONTROL/TESTSERV EXECUTES THE CODE
0000    199  ; FOLLOWING EACH NEXT TEST-CASE MACRO IMMEDIATELY BEFORE ISSUING
0000    200  ; THE SYSTEM SERVICE AS REQUESTED IN THE TESTSERV MACRO. TC CONTROL/
0000    201  ; TESTSERV ALSO CHECKS THE RESULTS OF THE SERVICE WITH RESPECT
0000    202  ; TO ITS EXPECTED STATUS CODE AND PRINTS ANY REQUIRED FAILURE
0000    203  ; MESSAGES FOR THE TEST CASE. THE CODE APPEARING AFTER EACH
0000    204  ; NEXT TEST CASE MACRO IS MERELY TO SET UP CONDITIONS REQUIRED
0000    205  ; FOR THE SYSTEM SERVICE AND TO CLEAN UP ANY RESOURCES ACQUIRED
0000    206  ; BY THE PREVIOUS TEST CASE.
0000    207  ;
0000    208  ; CALLING SEQUENCE:
0000    209  ;
0000    210  ;     $ RUN SATSSF08 ...  (DCL COMMAND)
0000    211  ;
0000    212  ; INPUT PARAMETERS:
0000    213  ;
0000    214  ;     NONE
0000    215  ;
0000    216  ; IMPLICIT INPUTS:
0000    217  ;
0000    218  ;     NONE
0000    219  ;
0000    220  ; OUTPUT PARAMETERS:
0000    221  ;
0000    222  ;     NONE
0000    223  ;
0000    224  ; IMPLICIT OUTPUTS:
0000    225  ;
0000    226  ;     MESSAGES TO SYS$OUTPUT ARE THE ONLY OUTPUT FROM SATSSF08.
0000    227  ;     THEY ARE OF THE FORM:
0000    228  ;
0000    229  ;             %UETP-S-SATSMS, TEST MODULE SATSSF08 BEGUN ... (BEGIN MSG)
0000    230  ;             %UETP-S-SATSMS, TEST MODULE SATSSF08 SUCCESSFUL ... (END MSG)
0000    231  ;             %UETP-E-SATSMS, TEST MODULE SATSSF08 FAILED ... (END MSG)
0000    232  ;             %UETP-I-TEXT, ... (VARIABLE INFORMATION ABOUT A TEST MODULE FAILURE)
0000    233  ;
0000    234  ; COMPLETION CODES:
0000    235  ;
0000    236  ;     THE SATSSF08 ROUTINE TERMINATES WITH A $EXIT TO THE
0000    237  ;     OPERATING SYSTEM WITH A STATUS CODE DEFINED BY UETP$_SATSMS.
0000    238  ;
0000    239  ; SIDE EFFECTS:
```

SATSSF08   C 4
V04-000   - SATS SYSTEM SERVICE TESTS (FAILING S. 16-SEP-1984 00:37:27 VAX/VMS Macro V04-00   Page 7
                    SATSSF08   5-SEP-1984 04:28:22 [UETPSY.SRC]SATSSF08.MAR;1   (1)

```
                              0000   240 ;
                              0000   241 ;         NONE
                              0000   242 ;
                              0000   243 ;--
                              0000   244
                              0000   245
                              0000   246
                              0000   247 SATSSF08:
                       OFFC   0000   248         .WORD    ^M<R2,R3,R4,R5,R6,R7,R8,R9,R10,R11>
                              0002   249                                                 ; ENTRY MASK
                              0002   250         $WAKE_S  TPID                           ; GET PID OF THIS PROCESS
                              0011   251         $HIBER_S                                ; UNDO WAKE
                              0018   252         $SETPRN_S TEST_MOD_NAME_D               ; SET PROCESS NAME
                    08C3  30  0025   253         BSBW     MOD_MSG_PRINT                  ; PRINT TEST MODULE BEGIN MSG
 00000060'EF  0000007D'EF DE  0028   254         MOVAL    TEST_MOD_SUCC,TMD_ADDR         ; ASSUME END MSG WILL SHOW SUCCESS
 00000044'EF  03  00  01  F0  0033   255         INSV     #SUCCESS,#0,#3,MOD_MSG_CODE    ; ADJUST STATUS CODE FOR SUCCESS
                              003C   256         MODE     TO,10$,KRNL,NOREGS             ; KERNEL MODE TO ACCESS PHD
        59  00000000'9F  DO  0059   257         MOVL     @#CTL$GL_PHD,R9                 ; GET PROCESS HEADER ADDRESS
           00000071'EF  69  DE  0060   258         MOVAL    PHD$Q_PRIVMSK(R9),PRIVMASK     ; GET PRIV MASK ADDRESS
                              0067   259         MODE     FROM,TO$                       ; GET BACK TO USER MODE
                              0068   260         PRIV     ADD,ALL                        ; GET ALL PRIVILEGES
                              0088   261         DISPSERV                                ; SET UP DISPLAY INFO FOR TESTSERV
                              021D   262         $SETPRT_S INADR=INADR, RETADR=RETADR, -
                              021D   263                   PROT=PROT, PRVPRT=PRVPRT
                              023E   264                                                 ; SET NOACCESS PSECT ...
                              023E   265                                                 ; ... FOR NO USER ACCESS
                    055B  31  023E   266         BRW      EXECUTE                        ; GO EXECUTE ALL TEST CASES
                              0241   267 ;
                              0241   268         TC_GROUP          CME,1,TS1
                              0268   269 ;
                              0268   270         NEXT_TEST_CASE  SFCME10
```

```
0268   271 ;
0268   272 ;++
0268   273 ; *****************************************************************
0268   274 ; *
0268   275 ; *   TEST CASE NAME:          SFCME10
0268   276 ; *
0268   277 ; *   SYSTEM SERVICE:          CMEXEC
0268   278 ; *
0268   279 ; *   ARGUMENT UNDER TEST:     ROUTIN_CME10
0268   280 ; *
0268   281 ; *   INPUT CONDITIONS:
0268   282 ; *       ISSUER OF CMEXEC SERVICE DOES NOT HAVE THE
0268   283 ; *       CHANGE MODE TO EXEC PRIVILEGE
0268   284 ; *
0268   285 ; *   EXPECTED RESULTS:
0268   286 ; *       1) SYSTEM STATUS CODE:  NOPRIV
0268   287 ; *       2) REGISTERS R2 THROUGH FP UNCHANGED
0268   288 ; *
0268   289 ; *****************************************************************
0268   290 ;--
0268   291 ;
0268   292         PRIV    REM,CMEXEC      ; REMOVE CHG MODE TO EXEC PRIVILEGE
0280   293         PRIV    REM,CMKRNL      ; REMOVE CHG MODE TO KERNEL PRIVILEGE
0298   294 ;
0298   295         TCEND
```

```
0299   296 ;
0299   297            TC_GROUP        GTM,1,TS2
02C0   298 ;
02C0   299            NEXT_TEST_CASE  SFGTM10
```

```
02C0   300 :
02C0   301 :++
02C0   302 : ***************************************************************
02C0   303 : *
02C0   304 : *  TEST CASE NAME:           SFGTM10
02C0   305 : *
02C0   306 : *  SYSTEM SERVICE:           GETMSG
02C0   307 : *
02C0   308 : *  ARGUMENT UNDER TEST:      MSGID_GTM10
02C0   309 : *
02C0   310 : *  INPUT CONDITIONS:
02C0   311 : *     MESSAGE IDENTIFICATION NOT IN USE BY SYSTEM
02C0   312 : *
02C0   313 : *  EXPECTED RESULTS:
02C0   314 : *     1) SYSTEM STATUS CODE:  MSGNOTFND
02C0   315 : *     2) REGISTERS R2 THROUGH FP UNCHANGED
02C0   316 : *
02C0   317 : ***************************************************************
02C0   318 :--
02C0   319 :
02C0   320        PRIV    ADD,ALL         ; GET BACK ANY PRIVS LOST BY LAST TEST CASE
02E0   321 :
02E0   322        NEXT_TEST_CASE  SFGTM30
```

SATSSF08
V04-000

G 4
- SATS SYSTEM SERVICE TESTS (FAILING S. 16-SEP-1984 00:37:27 VAX/VMS Macro V04-00   Page 11
SFGTM30   5-SEP-1984 04:28:22 [UETPSY.SRC]SATSSF08.MAR;1   (1)

```
02EC    323 ;
02EC    324 ;++
02EC    325 ; ***************************************************************
02EC    326 ; *
02EC    327 ; *   TEST CASE NAME:           SFGTM30
02EC    328 ; *
02EC    329 ; *   SYSTEM SERVICE:           GETMSG
02EC    330 ; *
02EC    331 ; *   ARGUMENT UNDER TEST:      BUFADR_GTM30
02EC    332 ; *
02EC    333 ; *   INPUT CONDITIONS:
02EC    334 ; *      MESSAGE BUFFER (LENGTH 0) NOT LARGE
02EC    335 ; *         ENOUGH TO CONTAIN MESSAGE.
02EC    336 ; *
02EC    337 ; *   EXPECTED RESULTS:
02EC    338 ; *      1) SYSTEM STATUS CODE:  BUFFEROVF
02EC    339 ; *      2) REGISTERS R2 THROUGH FP UNCHANGED
02EC    340 ; *
02EC    341 ; ***************************************************************
02EC    342 ;--
02EC    343 ;
02EC    344 ;
02EC    345          NEXT_TEST_CASE  SFGTM31
```

```
02F8   346
02F8   347 :.+
02F8   348 :.**********************************************************
02F8   349 :.*
02F8   350 :.*   TEST CASE NAME:         SFGTM31
02F8   351 :.*
02F8   352 :.*   SYSTEM SERVICE:         GETMSG
02F8   353 :.*
02F8   354 :.*   ARGUMENT UNDER TEST:    BUFADR_GTM31
02F8   355 :.*
02F8   356 :.*   INPUT CONDITIONS:
02F8   357 :.*      MESSAGE BUFFER (LENGTH 1) NOT LARGE
02F8   358 :.*      ENOUGH TO CONTAIN MESSAGE.
02F8   359 :.*
02F8   360 :.*   EXPECTED RESULTS:
02F8   361 :.*      1) SYSTEM STATUS CODE:  BUFFEROVF
02F8   362 :.*      2) REGISTERS R2 THROUGH FP UNCHANGED
02F8   363 :.*
02F8   364 :.**********************************************************
02F8   365 :.--
02F8   366 :.
02F8   367 :.
02F8   368          TCEND
```

```
02F9   369 TS1:
02F9   370          TESTSERV        CMEXEC,ERR,SATS,                                        -
02F9   371            <1,ROUTIN_CME,                                                        -
02F9   372                          ROUTIN_CME10,NOPRIV,    - ; SFCME10
02F9   373                                                            >,
02F9   374                                                                                  -
02F9   375            <1,ARGLST_CME,                                                        -
02F9   376                                                            >,                    -
02F9   377
02F9   378
0444   379          TS_CLEANUP              ; CLEAN UP & RETURN TO TEST_SERV_EXEC
```

```
0464   380 TS2:
0464   381          TESTSERV        GETMSG,ERR,SATS,                                        -
0464   382                                                                                  -
0464   383             <1,MSGID_GTM,                                                        -
0464   384                        MSGID_GTM10,MSGNOTFND,  -  ; SFGTM10
0464   385                                                    >,
0464   386                                                                                  -
0464   387             <1,MSGLEN_GTM,                                                       -
0464   388                                                    >,                            -
0464   389                                                                                  -
0464   390             <1,BUFADR_GTM,                                                       -
0464   391                        BUFADR_GTM30,BUFFEROVF,  -  ; SFGTM30
0464   392                        BUFADR_GTM31,BUFFEROVF,  -  ; SFGTM31
0464   393                                                    >,
0464   394                                                                                  -
0464   395             <1,FLAGS_GTM,                                                        -
0464   396                                                    >,                            -
0464   397                                                                                  -
0464   398             <1,OUTADR_GTM,                                                       -
0464   399                                                    >,                            -
0464   400
077C   401          TS_CLEANUP              ; CLEAN UP & RETURN TO TEST_SERV_EXEC
```

```
                              079C   402          .SBTTL  EXECUTE & CLEANUP
                              079C   403 EXECUTE:
                              079C   404          TEST_SERV_EXEC                      ; EXECUTE ALL T. CASES IN ALL GROUPS
                              07B0   405 CLEANUP:
                        0138  30 07B0  406          BSBW    MOD_MSG_PRINT             ; PRINT TEST MODULE END MSG
   00000044'EF  01  1C  01  F0 07B3  407          INSV    #1,#STS$V_INHIB_MSG,#1,MOD_MSG_CODE
                              07BC   408                                             ; INHIBIT PRINTING
                              07BC   409          $EXIT_S MOD_MSG_CODE               ; EXIT TO OP SYS WITH MSG CODE
```

```
07C9   411              .SBTTL  TC_CONTROL
07C9   412    ;++
07C9   413    ; FUNCTIONAL DESCRIPTION:
07C9   414    ;
07C9   415    ;        THE TC CONTROL SUBROUTINE IS CALLED BY THE TEST_SERV_EXEC
07C9   416    ; MACRO TO EXECUTE A GROUP OF TEST CASES. A GROUP IS DEFINED BY A TC-GROUP
07C9   417    ; MACRO. FOR EACH TC_GROUP MACRO, THERE IS A CORRESPONDING TESTSERV MACRO.
07C9   418    ; TESTSERV CONTAINS CODE TO EXECUTE SYSTEM SERVICES AND CHECK THE RETURNED
07C9   419    ; STATUS CODE VALUES. TESTSERV ARGUMENTS ARE CODED TO SPECIFY ALL THE SYSTEM
07C9   420    ; SERVICE ARGUMENT VALUES AND THE EXPECTED STATUS CODE FOR EACH TEST CASE
07C9   421    ; DEFINED BY A NEXT TEST CASE MACRO WITHIN THE GROUP. TC CONTROL USES A
07C9   422    ; CO-ROUTINE INTERFACE TO ENTER THE CODE OF THE APPROPRIATE TESTSERV MACRO
07C9   423    ; IN VARIOUS PLACES. THE FIRST ENTRY OCCURS ONCE PER GROUP TO ALLOW TESTSERV
07C9   424    ; TO DO SOME INITIALIZATION. THEN TWO ENTRIES ARE MADE FOR EACH TEST CASE IN
07C9   425    ; THE GROUP. THE FIRST ALLOWS TESTSERV TO ISSUE THE SUBJECT SYSTEM SERVICE.
07C9   426    ; THE SECOND ENTRY FOR THE TEST CASE CAUSES TESTSERV TO CHECK THE RETURNED
07C9   427    ; STATUS CODE, PRINTING A FAILURE MESSAGE IF IT IS NOT THE EXPECTED CODE.
07C9   428    ; IF THERE ARE NO MORE TEST CASES IN THE CURRENT GROUP, TESTSERV (NOT TC CONTROL)
07C9   429    ; RETURNS DIRECTLY TO TEST_SERV_EXEC (RSB ACTUALLY ISSUED IN TS CLEANUP MACRO)
07C9   430    ; FROM THIS SECOND ENTRY; OTHERWISE, CONTROL RETURNS TO TC CONTROL WHICH
07C9   431    ; IN TURN ENTERS TESTSERV AGAIN FOR THE NEXT TEST CASE. THE FAILURE OF A
07C9   432    ; TEST CASE DOES NOT CAUSE TERMINATION OF THE TEST MODULE.
07C9   433    ;
07C9   434    ; CALLING SEQUENCE:
07C9   435    ;
07C9   436    ;        BSBW TC_CONTROL (ISSUED WITHIN THE TEST_SERV_EXEC MACRO)
07C9   437    ;                        (RSB IS ISSUED WITHIN THE TS_CLEANUP MACRO)
07C9   438    ;
07C9   439    ; INPUT PARAMETERS:
07C9   440    ;
07C9   441    ;        NONE
07C9   442    ;
07C9   443    ; IMPLICIT INPUTS:
07C9   444    ;
07C9   445    ;        ARGUMENTS SPECIFIED ON EACH TESTSERV MACRO MAY BE VIEWED AS
07C9   446    ;        INPUTS, SINCE TC_CONTROL AND TESTSERV ACT AS CO-ROUTINES.
07C9   447    ;
07C9   448    ; OUTPUT PARAMETERS:
07C9   449    ;
07C9   450    ;        SEVERITY CODE FIELD OF MOD_MSG_CODE (BITS 0,1,2) IS SET TO ERROR
07C9   451    ;        IF ANY TEST CASE IN THE CURRENT GROUP FAILS; OTHERWISE IT REMAINS
07C9   452    ;        SET TO SUCCESSFUL.
07C9   453    ;
07C9   454    ; IMPLICIT OUTPUTS:
07C9   455    ;
07C9   456    ;        %UETP-I-TEXT,   ERROR MESSAGES ARE WRITTEN TO SYS$OUTPUT BY
07C9   457    ;                        THE TESTSERV MACRO (CO-ROUTINE WITH TC_CONTROL)
07C9   458    ;
07C9   459    ; COMPLETION CODES:
07C9   460    ;
07C9   461    ;        NONE
07C9   462    ;
07C9   463    ; SIDE EFFECTS:
07C9   464    ;
07C9   465    ;        NONE
07C9   466    ;
07C9   467    ;--
```

SATSSF08
V04-000
```
                        M 4
      - SATS SYSTEM SERVICE TESTS  (FAILING S.  16-SEP-1984 00:37:27   VAX/VMS Macro V04-00        Page  17.
        TC_CONTROL                                5-SEP-1984 04:28:22   [UETPSY.SRC]SATSSF08.MAR;1        (1)
```

```
                              07C9    468
                              07C9    469
                              07C9    470
                              07C9    471   TC_CONTROL:
        00000064'EF   DD      07C9    472        PUSHL    TS_EP                    ; PUSH TESTSERV ENTRY POINT
                9E   16      07CF    473        JSB      @(SP)+                   ; ENTER TESTSERV INITIALIZATION
                              07D1    474   10$:                                   ; PROCESS NEXT TEST CASE
        00000056'EF   20  90  07D1    475        MOVB     #^A/ /,.S$TSTN$$+2       ; MAKE SURE T.C. NAME HAS A BLANK
                002F   30      07D8    476        BSBW     REG_SAVE                 ; SAVE REGISTERS
        00000004'FF   16      07DB    477        JSB      @CURRENT_TC              ; JUMP TO CURRENT TEST CASE
                0037   30      07E1    478        BSBW     REG_REST                 ; RESTORE REGS FOR TESTSERV
                9E   16      07E4    479        JSB      @(SP)+                   ; LET TESTSERV ISSUE SYSTEM SERVICE
                0042   30      07E6    480        BSBW     REG_COMP                 ; COMPARE REGS TO SEE IF ...
                              07E9    481                                         ; ... SYSTEM SERVICE CHANGED ANY
                9E   16      07E9    482        JSB      @(SP)+                   ; LET TESTSERV CHEK S.S. STATUS CODE
        00000056'EF   2A  91  07EB    483        CMPB     #^A/*/,.S$TSTN$$+2       ; HAS TESTSERV INDICATED FAILURE ?
                DD   12      07F2    484        BNEQU    10$                      ; NO -- PROCESS NEXT TEST CASE
00000060'EF  00000088'EF  DE  07F4    485        MOVAL    TEST_MOD_FAIL,TMD_ADDR   ; YES -- INDICATE FAILED IN END MSG
00000044'EF  03  00  02  F0  07FF    486        INSV     #ERROR,#0,#3,MOD_MSG_CODE ; ADJUST STATUS CODE FOR ERROR
                C7   11      0808    487        BRB      10$                      ; LOOP BAK TO PROCESS NEXT TEST CASE
                              080A    488   ;
                              080A    489   ; TC_CONTROL RETURNS TO TEST_SERV_EXEC VIA TESTSERV (IN TS_CLEANUP MACRO)
                              080A    490   ;
```

```
                         080A        492                 .SBTTL   SUBROUTINES
                         080A        493  REG_SAVE:
                         080A        494  ;
                         080A        495  ;    **************************************************************
                         080A        496  ;    *                                                            *
                         080A        497  ;    *   SAVES R0 THRU SP IN REG_SAVE_AREA                         *
                         080A        498  ;    *                                                            *
                         080A        499  ;    **************************************************************
                         080A        500  ;
              7FFF 8F  BB  080A       501                 PUSHR    #R0_THRU_SP             ; SAVE ALL REGS ON STACK
   00000008'EF  6E  3C  28  080E      502                 MOVC3    #60,(SP),REG_SAVE_AREA  ; SAVE REGS (BEFORE S.S.)
              7FFF 8F  BA  0816       503                 POPR     #R0_THRU_SP             ; CLEAN UP STACK
                        05  081A      504                 RSB                             ; .... AND RETURN
                         081B        505  ;
                         081B        506  ;
                         081B        507  ;
                         081B        508  ;
                         081B        509  REG_REST:
                         081B        510  ;
                         081B        511  ;
                         081B        512  ;    **************************************************************
                         081B        513  ;    *                                                            *
                         081B        514  ;    *   RESTORES R0 THRU SP FROM REG_SAVE_AREA                    *
                         081B        515  ;    *                                                            *
                         081B        516  ;    **************************************************************
                         081B        517  ;
              5E  3C  C2  081B        518                 SUBL2    #60,SP                  ; MOVE SP TO MAKE ROOM FOR REGS
   6E  00000008'EF  3C  28  081E     519                 MOVC3    #60,REG_SAVE_AREA,(SP)  ; MOVE REGS ONTO STACK FOR POP
              7FFF 8F  BA  0826       520                 POPR     #R0_THRU_SP             ; RESTORE ALL REGS FOR TESTSERV
                        05  082A      521                 RSB                             ; ... AND RETURN
```

```
                                    082B     523  REG_COMP:
                                    082B     524  ;
                                    082B     525  ;    ***************************************************************
                                    082B     526  ;    *                                                             *
                                    082B     527  ;    *   1) PUSHES ALL REGS ONTO STACK                             *
                                    082B     528  ;    *   2) COMPARES REGISTER IMAGES FROM STACK WITH CORRESPONDING *
                                    082B     529  ;    *         IMAGES FROM REG_SAVE_AREA FOR ALL REGISTERS SPECIFIED*
                                    082B     530  ;    *         IN REG_COMP_MASK.                                   *
                                    082B     531  ;    *   3) FOR EACH UNEQUAL COMPARE, AN ERROR MESSAGE IS PRINTED  *
                                    082B     532  ;    *         (USING $FAO AND $OUTPUT SYSTEM SERVICES).           *
                                    082B     533  ;    *   4) POPS ALL REGS OFF OF STACK                             *
                                    082B     534  ;    *                                                             *
                                    082B     535  ;    ***************************************************************
                                    082B     536  ;
                      7FFF 8F   BB  082B     537          PUSHR   #R0_THRU_SP                 ; SAVE ALL REGISTERS ON STACK
        56    00000008'EF   DE  082F     538          MOVAL   REG_SAVE_AREA,R6            ; POINT R6 TO BEG OF ...
                                    0836     539                                          ; ... REGS (BEFORE S.S.)
                      54    5E   D0  0836     540          MOVL    SP,R4                      ; POINT R4 TO BEG OF
                                    0839     541                                          ; ... REGS (AFTER S.S.)
              53    FF 8F   98  0839     542          CVTBL   #-1,R3                     ; INITIALIZE REG_COMP_MASK INDEX
                                    083D     543  REG_COMP_NEXT:
                            53   D6  083D     544          INCL    R3                         ; POINT TO NEXT BIT IN MASK
                      53    0F   91  083F     545          CMPB    #15,R3                     ; END OF THE MASK ?
                            03   1A  0842     546          BGTRU   REG_COMP_CONT              ; NO -- CONTINUE
                          009F   31  0844     547          BRW     REG_COMP_RSB               ; YES -- GO TO COMMON RETURN
                                    0847     548  REG_COMP_CONT:
                      84    86   D1  0847     549          CMPL    (R6)+,(R4)+                ; REG BEFORE = REG AFTER ?
                            F1   13  084A     550          BEQLU   REG_COMP_NEXT              ; YES -- LOOK FOR NEXT REG
      E9  00000000'EF   53   E1  084C     551          BBC     R3,REG_COMP_MASK,REG_COMP_NEXT
                                    0854     552                                          ; NO -- GET NEXT IF BIT NOT SET
          00000048'EF   53   D0  0854     553          MOVL    R3,CLOB_REG_NO             ; NO -- GIVE REG NUMBER TO FAO
      0000004C'EF   FC A6   D0  085B     554          MOVL    -4(R6),REG_BEFORE_SS      ; GIVE "BEFORE" CONTENTS TO FAO
      00000050'EF   FC A4   D0  0863     555          MOVL    -4(R4),REG_AFTER_SS       ; GIVE "AFTER" CONTENTS TO FAO
          00000056'EF   2A   90  086B     556          MOVB    #^A/*/,$$TSTN$$+2         ; GIVE FAILURE INDIC'N IN ERROR MSG
                                    0872     557  ;
                                    0872     558          $FAO_S  ERR_MSG_FAOCTL,OUTL,OUTD,$$SNAD$$, -
                                    0872     559                  $$ASEQ$$,$$PSEQ$$,CLOB_REG_NO,REG_BEFORE_SS,REG_AFTER_SS
                                    08A5     560  ;
      F868 CF   F832 CF   B0  08A5     561          MOVW    OUTL,OUTD                  ; ACTUAL OUTPUT LEN IN STRING DESC'R
                                    08AC     562          PUTMSG  <#UETP$_TEXT,#1,#OUTD>     ; PRINT THE MSG
      F84C CF   0084 8F   B0  08C1     563          MOVW    #OUTE-OOTB,OUTD            ; GET MAX LEN BACK INTO DESCRIPTOR
          00000056'EF   20   90  08C8     564          MOVB    #^A/ /,$$TSTN$$+2         ; REMOVE FAIL INDIC'N FOR NEXT MSG
      00000060'EF   00000088'EF   DE  08CF     565          MOVAL   TEST_MOD_FAIL,TMD_ADDR    ; INDICATE FAILED IN END MSG
      00000044'EF   03  00   02  F0  08DA     566          INSV    #ERROR,#0,#3,MOD_MSG_CODE ; ADJUST STATUS CODE FOR ERROR
                          FF57   31  08E3     567          BRW     REG_COMP_NEXT              ; GO LOOK FOR NEXT REG TO COMPARE
                                    08E6     568  REG_COMP_RSB:
                      7FFF 8F   BA  08E6     569          POPR    #R0_THRU_SP                ; CLEAN UP STACK
                            05  08EA     570          RSB                                ; RETURN TO CALLER
```

```
                    08EB   572 MOD_MSG_PRINT:
                    08EB   573 ;
                    08EB   574 ; ****************************************************************
                    08EB   575 ;  *                                                              *
                    08EB   576 ;  *   PRINTS THE TEST MODULE BEGUN/SUCCESSFUL/FAILED MESSAGES     *
                    08EB   577 ;  *        (USING THE PUTMSG MACRO).                              *
                    08EB   578 ;  *                                                              *
                    08EB   579 ; ****************************************************************
                    08EB   580 ;
                    08EB   581        PUTMSG  <MOD_MSG_CODE,#2,TMN_ADDR,TMD_ADDR> ; PRINT MSG
                05  0906   582        RSB                              ; ... AND RETURN TO CALLER
                    0907   583 ;
                    0907   584 CHMRTN:
                    0907   585 ; ****************************************************************
                    0907   586 ;  *                                                              *
                    0907   587 ;  *   CHANGE MODE ROUTINE. THIS ROUTINE GETS CONTROL WHENEVER    *
                    0907   588 ;  *   A CMKRNL, CMEXEC, OR CMSUP SYSTEM SERVICE IS ISSUED         *
                    0907   589 ;  *   BY THE MODE MACRO ('TO' OPTION). IT MERELY DOES             *
                    0907   590 ;  *   A JUMP INDIRECT ON A FIELD SET UP BY MODE. IT HAS           *
                    0907   591 ;  *   THE EFFECT OF RETURNING TO THE END OF THE MODE              *
                    0907   592 ;  *   MACRO EXPANSION.                                           *
                    0907   593 ;  *                                                              *
                    0907   594 ; ****************************************************************
                    0907   595 ;
               0000 0907   596        .WORD   0                        ; ENTRY MASK
  00000079'FF    17 0909   597        JMP     @CHM_CONT                ; RETURN TO MODE MACRO IN NEW MODE
                    090F   598 ;
                    090F   599 ; *    RET INSTR WILL BE ISSUED IN EXPANSION OF 'MODE FROM, .....' MACRO
                    090F   600 ;
                    090F   601        .END    SATSSF08
```

Symbol	Value			Symbol	Value		
$$$CHARS	= 00000048			PRV$V_CMKRNL	= 00000000		
$$$FIRSTTC$$$	= 00000000			PRVPRT	00000070	R	03
$$$STRINGS	= 00000000			R0_THRU_SP	= 00007FFF		
$$ACT$$	000000F3	R	06	REGS	0000007D	R	03
$$ARG$$	000000FB	R	06	REG_AFTER_SS	00000050	R	03
$$ASEQ$$	000000EB	R	06	REG_BEFORE_SS	0000004C	R	03
$$CALL$$	000000DF	R	06	REG_COMP	0000082B	R	06
$$DISP$$	000001E6	R	06	REG_COMP_CONT	00000847	R	06
$$ERR$$	000001A0	R	06	REG_COMP_MASK	00000000	R	02
$$EXP$$	000000F7	R	06	REG_COMP_NEXT	0000083D	R	06
$$INIT$$	000000E3	R	06	REG_COMP_RSB	000008E6	R	06
$$MAXP$$	= 00000005			REG_REST	0000081B	R	06
$$PSEQ$$	000000EF	R	06	REG_SAVE	0000080A	R	06
$$SNAD$$	000000E7	R	06	REG_SAVE_AREA	00000008	R	03
$$T1	= 00000004			RETADR	00000068	R	03
$$T2	= 00000009			ROUTIN_CME	000000BD	R	02
$$TSTN$$	00000054	R	03	ROUTIN_CME10	000000BD	R	02
ARGLST_CME	= 00000000			SATSSF08	00000000	R	06
BUFADR_GTM	00000093	R	03	SEVERE	= 00000004		
BUFADR_GTM30	0000019F	R	03	SHR$K_SHRDEF	= 00000001		
BUFADR_GTM31	000001A7	R	03	SHR$_TEXT	= 00001130		
CHMRTN	00000907	R	06	SS$_BUFFEROVF	********	X	06
CHM_CONT	00000079	R	03	SS$_MSGNOTFND	********	X	06
CLEANUP	000007B0	R	06	SS$_NOPRIV	********	X	06
CLOB_REG_NO	00000048	R	03	SS$_NORMAL	********	X	02
CTL$GL_PRD	********	X	06	STS$V_INHIB_MSG	= 0000001C		
CURRENT_TC	00000004	R	03	SUCCESS	= 00000001		
EMPTY	00000000	R	04	SYS$CMEXEC	********	GX	06
ERROR	= 00000002			SYS$CMKRNL	********	GX	06
ERR_MSG_FAOCTL	00000002	R	02	SYS$EXIT	********	GX	06
EXECUTE	0000079C	R	06	SYS$FAO	********	X	06
FLAGS_GTM	000000C8	R	02	SYS$FAOL	********	GX	06
GRP_TOTAL	= 00000002			SYS$GETMSG	********	GX	06
INADR	000000A9	R	02	SYS$HIBER	********	GX	06
INFO	= 00000003			SYS$SETPRN	********	GX	06
LIB$SIGNAL	********	X	06	SYS$SETPRT	********	GX	06
MEXIT	= 00000000			SYS$SETPRV	********	GX	06
MOD_MSG_CODE	00000044	R	03	SYS$WAKE	********	GX	06
MOD_MSG_PRINT	000008EB	R	06	TC1	00000241	R	06
MSGID_GTM	000000C4	R	02	TC2	00000299	R	06
MSGID_GTM10	000000CC	R	02	TCG_NO	= 00000002		
MSGLEN_GTM	00000091	R	03	TC_CONTROL	000007C9	R	06
NARGS	= 0000000E			TEST_MOD_BEG	00000077	R	02
NOACCESS	00000000	R	05	TEST_MOD_FAIL	00000088	R	02
NSSARGS	= 00000005			TEST_MOD_NAME	0000006E	R	02
ONES	000000B5	R	02	TEST_MOD_NAME_D	0000008F	R	02
OUTADR_GTM	0000019B	R	03	TEST_MOD_SUCC	0000007D	R	02
OUTB	0000011C	R	06	TMD_ADDR	00000060	R	03
OUTD	00000114	R	06	TMN_ADDR	0000005C	R	03
OUTE	000001A0	R	06	TPID	00000000	R	03
OUTL	000000DB	R	06	TS1	000002F9	R	06
PHD$Q_PRIVMSK	= 00000000			TS2	00000464	R	06
PRIVMASK	00000071	R	03	TS_EP	00000064	R	03
PRIV_ARGS	= 00000002			TTNAME	0000009F	R	02
PROT	000000B1	R	02	UETP$_SATSMS	= 007480D9		
PRT$C_NA	********	X	02	UETP$_TEXT	= 00741133		
PRV$V_CMEXEC	= 00000001			WARNING	= 00000000		

E 5

```
                                    +-------------------+
                                    ! Psect synopsis !
                                    +-------------------+

PSECT name                  Allocation          PSECT No.   Attributes
-----------                 ----------          ---------   ----------
.   ABS   .                 00000000  (     0.)  00 (  0.)  NOPIC  USR  CON  ABS  LCL  NOSHR  NOEXE  NORD  NOWRT  NOVEC  BYTE
$ABS$                       00000000  (     0.)  01 (  1.)  NOPIC  USR  CON  ABS  LCL  NOSHR   EXE    RD   WRT    NOVEC  BYTE
RODATA                      000000D0  (   208.)  02 (  2.)  NOPIC  USR  CON  REL  LCL  NOSHR  NOEXE   RD   NOWRT  NOVEC  LONG
RWDATA                      000001B0  (   432.)  03 (  3.)  NOPIC  USR  CON  REL  LCL  NOSHR  NOEXE   RD   WRT    NOVEC  BYTE
SATS_ACCVIO_1               00000200  (   512.)  04 (  4.)  NOPIC  USR  CON  REL  LCL  NOSHR  NOEXE   RD   WRT    NOVEC  PAGE
SATS_ACCVIO_2               00000200  (   512.)  05 (  5.)  NOPIC  USR  CON  REL  LCL  NOSHR  NOEXE   RD   WRT    NOVEC  PAGE
SATSSF08                    0000090F  (  2319.)  06 (  6.)  NOPIC  USR  CON  REL  LCL  NOSHR   EXE    RD   WRT    NOVEC  LONG
```

```
                              +-----------------------------+
                              ! Performance indicators !
                              +-----------------------------+

Phase                     Page faults    CPU Time        Elapsed Time
-----                     -----------    --------        ------------
Initialization                 38        00:00:00.07     00:00:00.39
Command processing            141        00:00:00.60     00:00:02.45
Pass 1                        339        00:00:12.70     00:00:27.30
Symbol table sort               0        00:00:00.97     00:00:01.98
Pass 2                        128        00:00:02.76     00:00:05.62
Symbol table output            14        00:00:00.10     00:00:00.30
Psect synopsis output           3        00:00:00.03     00:00:00.04
Cross-reference output          0        00:00:00.00     00:00:00.00
Assembler run totals          665        00:00:17.23     00:00:38.08
```

The working set limit was 1650 pages.
66336 bytes (130 pages) of virtual memory were used to buffer the intermediate code.
There were 40 pages of symbol table space allocated to hold 593 non-local and 78 local symbols.
601 source lines were read in Pass 1, producing 26 object records in Pass 2.
62 pages of virtual memory were used to define 46 macros.

```
                              +------------------------------+
                              ! Macro library statistics !
                              +------------------------------+

Macro library name                      Macros defined
------------------                      --------------
_$255$DUA28:[SHRLIB]UETP.MLB;1                19
_$255$DUA28:[SYS.OBJ]LIB.MLB;1                 2
_$255$DUA28:[SYSLIB]STARLET.MLB;2             19
TOTALS (all libraries)                        40
```

1226 GETS were required to define 40 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS$:SATSSF08/OBJ=OBJ$:SATSSF08 MSRC$:SATSSF08/UPDATE=(ENH$:SATSSF08)+EXECML$/LIB+SHRLIB$:UETP/LIB